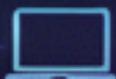


TechPads



Manual de Configuración

Carrito de compras online



CONTENIDO

1.	Introducción.....	2
2.	Configuración de Entorno	2
2.1.	Configuración General del Sistema	2
3.	Configuración FrontEnd	3
3.1.	Configuracion Variables de Entorno.....	3
4.	Configuración BackEnd	3
4.1.	Configuración de la Conexión de la Base de Datos	3
4.2.	Configuración del Puerto y Dirección del Path	4
5.	Instalación de la Base de Datos en Oracle	4
6.	Configuración de Seguridad	8
6.1.	Autenticación Mediante JWT	8
7.	Roles y Permisos	9
8.	Pruebas y Verificación	9
8.1.	Pruebas de FrontEnd	9
8.2.	Pruebas de Back	10
8.3.	Prueba de Bases de Datos	10

Manual de Configuración

1. Introducción

Este manual describe los ajustes necesarios para que el sistema *delivery-shop*, compuesto por un frontend desarrollado en Angular, un backend construido en Spring Boot y una base de datos Oracle, quede correctamente configurado en un entorno local o de producción.

Mientras que el manual de instalación indica cómo instalar los componentes, este manual explica cómo configurarlos para que funcionen de forma integrada.

2. Configuración del Entorno

2.1. Configuración general del sistema

Antes de configurar los módulos, se deben verificar los siguientes elementos:

- Java 17 o superior configurado en la variable JAVA_HOME.
- Node.js 18 o superior.
- Angular CLI instalado globalmente.
- Base de datos Oracle 19c o superior funcionando correctamente.
- Acceso al repositorio del proyecto.
- Puertos disponibles: 4200 (frontend) y 8080 (backend).

3. Configuración del Frontend (Angular)

3.1. Configuración de variables de Entorno

Entrar en el frontend, en los environments y debe estar configurado a la dirección del back

```
TypeScript
export const environment = {
  production: false,
  apiUrl: 'http://localhost:8080/techPads/store/v1'
};
```

Siendo <http://localhost:8080/techPads/store/v1> el predeterminado de la página

4. Configuración del Backend (Spring boot)

4.1. Configuración de la conexión a la base de datos

Entrar al Backend, en application.properties deben estar todas las configuraciones iniciales, como la conexión de la base de datos

```
Java
spring.datasource.url=jdbc:oracle:thin:@localhost:1521:XE
spring.datasource.username=TechPads
spring.datasource.password=oracle
spring.datasource.driver-class-name=oracle.jdbc.OracleDriver

#Configuracion JPA / Hibernate
spring.jpa.hibernate.ddl-auto=validate
spring.jpa.show-sql=true
spring.jpa.database-platform=org.hibernate.dialect.OracleDialect
spring.jpa.hibernate.naming.physical-strategy=org.hibernate.boot.model.naming.PhysicalNamingStrategyStandardImpl
```

En el que aparece como base de datos TechPads con la contraseña Oracle, si se desea cambiar el usuario o la base de datos de oracle a otros servicio se debe hacer cambiado esta parte de código.

4.2. Configuración del puerto y dirección del path

Por defecto Spring Boot usa el puerto 8080, pero puede modificarse:

Java

```
server.port=8080
```

Si se desea colocar el back en otro puerto se puede cambiar en esta parte

Nota: Al cambiar el puerto (8080) a otro diferente, se tendrá que tener en cuenta cambios en el front y la dirección del back

El path viene como predeterminado

Java

```
server.servlet.context-path=/techPads/store/v1
```

Si se desea cambiar por otro se hará en esta línea de código

Nota: Al cambiar la dirección del path se tiene que tener en cuenta cambios en el front con las direcciones.

5. Instalación a la base de datos en Oracle

Sugerencia: Para la creación de la base de datos necesitamos crear un usuario o base de datos con el nombre de **TechPads** con la contraseña de **Oracle**

En el repositorio de git aparece la carpeta de sql por lo que ejecutaremos los scripts en el siguiente orden:

1. DDL.sql
2. FUNCTIONS_AND PROCEDURES.sql

3. TRIGGERS.sql
4. INDICES-DICCIONARIO-PRUEBAS.sql (parte de índices)
5. INSERTS (opcional → datos de prueba)

Además de ello, los errores que puede arrojar la base de datos por parte de paquetes de pl/sql estan listados de la siguiente manera:

Código de la excepción	Descripción	Tabla afectada
-20101	El producto ya existe	PRODUCT
-20102	Error al crear el producto	PRODUCT
-20103	No existe un producto con el código indicado	PRODUCT
-20104	Error al modificar el producto	PRODUCT
-20105	Error al desactivar el producto y su inventario	PRODUCT / INVENTORY
-20106	Producto no encontrado	PRODUCT
-20107	Error al consultar producto	PRODUCT
-20108	Error al listar productos	PRODUCT
-20109	Error al cargar productos activos en la colección	PRODUCT
-20201	El código de inventario ya existe	INVENTORY
-20202	Error al registrar el lote	INVENTORY
-20203	No existe un lote con ese código	INVENTORY
-20204	Error al modificar el inventario	INVENTORY
-20205	Inventario no encontrado	INVENTORY

-20206	Error al consultar inventario	INVENTORY
-20207	Error al verificar el stock	INVENTORY
-20301	El nombre de usuario o correo ya existe	USERS / CLIENT_DETAIL
-20302	Error registrando usuario	USERS / CLIENT_DETAIL
-20303	El usuario no existe	USERS
-20304	Error modificando usuario	USERS / CLIENT_DETAIL
-20305	No existe un usuario con el ID proporcionado	USERS / CLIENT_DETAIL
-20306	Error consultando usuario	USERS / CLIENT_DETAIL
-20401	El producto no existe o no está disponible	PRODUCT
-20402	Stock insuficiente	INVENTORY
-20403	Error al agregar al carrito	CART
-20404	Stock insuficiente	INVENTORY
-20405	Carrito no encontrado	CART
-20406	Error al actualizar cantidad	CART
-20407	Carrito no encontrado	CART
-20408	Error al eliminar del carrito	CART
-20409	Error al limpiar carrito	CART
-20410	Error al consultar carrito	CART / PRODUCT
-20411	Error al calcular total	CART / PRODUCT

-20412	Error al verificar disponibilidad	INVENTORY
-20501	El carrito está vacío	CART
-20502	Stock insuficiente para uno o más productos	INVENTORY
-20503	Error al procesar la compra	ORDER / ORDER_DETAIL / BILL / INVENTORY
-20504	No se puede cancelar una orden entregada o ya cancelada	ORDER
-20505	Orden no encontrada	ORDER
-20506	Error al cancelar orden	ORDER / INVENTORY
-20507	Orden no encontrada	ORDER
-20508	Error al cambiar estado	ORDER
-20509	Error al consultar órdenes	ORDER / BILL
-20510	Orden no encontrada	ORDER
-20511	Error al consultar detalle	ORDER_DETAIL / PRODUCT

6. Configuración de Seguridad

La seguridad del sistema *delivery-shop* se basa en un esquema de autenticación mediante JWT (JSON Web Tokens). Este método permite controlar el acceso a los recursos del backend garantizando que cada petición provenga de un usuario previamente autenticado. A continuación se detalla cómo funciona este mecanismo y cómo se configura dentro del proyecto.

6.1. Autenticación mediante JWT

El sistema divide el proceso de seguridad en dos etapas principales:

1. Autenticación inicial

- El usuario envía sus credenciales (correo/usuario y contraseña) al backend.
- Spring Boot verifica la información usando el servicio de autenticación.
- Si los datos son correctos, el backend genera un **token JWT** firmado con una clave secreta.
- Ese token se devuelve al frontend y será incluido en cada solicitud posterior.

2. Autorización en cada petición

- Cada vez que el usuario accede a un recurso protegido, el frontend envía el token en la cabecera HTTP:

`Authorization: Bearer <token>`

- Spring Boot valida:
 - Que el token no esté alterado.
 - Que no esté vencido.
 - Que pertenezca a un usuario válido.
- Si la validación es correcta, se permite el acceso al recurso solicitado.

7. Roles y permisos

En la aplicación como tal se tiene definidos dos roles los cuales son y son capaces de:

Administrador:

- crear, modificar, desactivar y consultar productos
- crear, modificar, desactivar y consultar lotes (inventario)
- modificar y visualizar la gestión de órdenes

Cliente:

- consultar productos
- consultar inventario
- añadir productos al carrito
- comprar productos
- generar órdenes
- modificar datos del usuario

8. Pruebas de Verificación

8.1. Prueba del Frontend

- Navegar entre componentes.
- Revisar funcionamiento de formularios.
- Tomar tiempos de carga.

8.2. Prueba del Backend

Acceder directamente a un endpoint: <http://localhost:8080/techPads/store/v1>

- Verificar respuestas JSON.

Sugerencia: para las pruebas de los diferentes endpoints del back se sugiere hacerlo en la app de Postman

8.3. Prueba de Base de Datos

- Comprobar inserciones y consultas.
- Revisar los logs del backend.